Proceedings of the International Conference on Inventive Research in Computing Applications (ICIRCA 2022)
IEEE Xplore Part Number: CFP22N67-ART; ISBN: 978-1-6654-9707-7

Movie Recommendation System using Machine Learning

Narendra Kumar Rao[1]
Department of Computer Science and Engineering
Sree Vidyanikethan Engineering College
Tirupathi, India
narendrakumarraob@gmail.com

Nagendra Panini Challa[2]
School of Computer Science and Engineering
VIT-AP University, Amaravati
Andhra Pradesh, India
paninichalla123@gmail.com

S Sreenivasa Chakravarthi[3]
Department of CSE, Amrita School of Computing,
Amrita Viswa Vidyapeetham, Chennai, India
ss_chakravarthi@ch.amrita.edu

Ranjana R[4]
Department of Information Technology,
Sri Sairam Engineering College, Chennai, India
ranjana.it@sairam.edu.in

2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA) | 978-1-6654-9707-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICIRCA54612.2022.9985512

***Abstract*— The recommendation engine filters information using specific algorithms and recommends high quality content to customers. It starts capturing more consumer behavior and based on that, recommends products that consumers can purchase. Three key strategies are used in our recommendation structures. One Demographic Filtering i.e. They offer general suggestions for each individual, based entirely on the film's image and genre. The system recommends similar films to all the users. if you consider that each person is of the same type, this method is considered very simple. The simple idea behind is that the movies which are more popular can be liked by the more people. The second method is content based filtering, which considers all the features like director, actors and movie related content and based on that the movies will be recommended. The third one is collaborative filtering, which implement the item based collaborative filtering and single value decomposition. The obtained results have showcased the proposed strategies with good accuracy.**

***Keywords*— *Demographic filtering, Collaborative based filtering, content-based filtering, recommendation system (RS), Machine Learning.*

I. INTRODUCTION

A recommendation engine is a kind of filtering which filters the content and make suggestions to the users based on their preferences. There is an extensive type of packages for recommendation systems. Due to increase in the applications and online e-commerce applications the recommendation system plays a crucial role. The content of such structures varies from films, song, books and films, to buddies and memories on social media structures, to products rate web sites, to people on professional and courting web sites, to look outcomes again on Google [1]. regularly, those systems are able to acquire information about a customer's alternatives, and may use these records to enhance their tips inside the destiny. As an example, fb can screen your interplay with diverse memories on your feed a good way to learn what forms of stories enhancement to you. on occasion, the recommender systems could make improvements based at the activities of a massive wide variety of people. For example, take any online store if one buys a mobile phone there is a high chance of buying an earphone by that user so it will recommend to that user. Due to many advances in recommender structures, customers continuously assume top tips. If a video streaming app is not capable of are expecting and play track that the user likes, then the consumer will virtually prevent using it. This has brought about a high emphasis by using tech companies on enhancing their recommendation structures [2]. But the trouble is greater complicated than it appears. Each user has one-of-a-kind options and likes. for instance, the sort of song one would like to pay attention at the same time as exercising differs substantially from the kind of song he'd listen to while doing some work. They ought to explore new domain names to find out greater approximately the user. Some of the filtering methods are applied in the recommender systems. First, the demographic-filtering is applied to the system for recommending same movies to all the users. The device recommends the same films to users with comparable demographic functions [3]. The simple concept in the back of this system is that films which can be more famous and is more likely to be preferred by most of the audience. Later the content-based filtering is applied by considering all the features like director, actors and movie related content and based on that the movies will be recommended. The third one is collaborative filtering, in which an item based collaborative filtering and single value decomposition is applied based on the movie ratings given by the user. The primary objective of this project is to recommend a movie to the user. The major goal of this project is to predict the movie rating given by a particular user in which the user is not previously rated [4]. This can be performed by using demographic filtering, content-based filtering, collaborative-based filtering. There is wide range of scope for this recommender system because every e-commerce website uses this recommendation system without this their usage may be very less. Due to their scalability and accuracy, it has wide range of scope, but there are still many problems related to quality and scalability in recommender systems [5].

II. LITERATURE SURVEY

Sudhanshu kumar proposed a "movie recommendation system using sentiment analysis from microblogging data" in which he had used hybrid system which is a combination of content-based filtering and collaborative-based filtering with the sentiment analysis which is done through taking tweets from several micro blogging sites [6]. So tha data taken from public tweet will give the better results. MOVREC is a movie recommendation system offered by way of Mukesh Kumar Karitha based on collaborative-based filtering approach. collaborative-based filtering will use the similarities between the users and items and recommend movies to the user. The data of the datasets are analysed and recommends movies to the user with the highest rating [7].

Harpreet Kaur et al has proposed the hybrid system. This hybrid system is a combination of content-based filtering and collaborative-based filtering [8]. This is based on content-based filtering in which it will consider the contents like director, actor, genres of the movie and recommend the similar movies. In collaborative-based filtering which uses user user relationship and user item relationship. The user data and item data is combined to form a cluster which is based on green approach.This system has lower errors and it clusters the similar items in a better way. To calculate the score numbers of votes are considered in this system. Luis M Capos et al has analysed many recommender systems like content-based filtering and collaborative based filtering as both of them has their own disadvantages he proposed a new recommender system by combining collaborative-based filtering and Bayesian network.

Hongli LIn et had proposed a system in which first he will apply content-based filtering totally to the data and boost this system again by applying collaborative filtering to the system so that it can had the benefits of both content-based filtering and collaborative-based filtering. First it will improve the contents of the system and then it improves the quality by the collaborative-filtering in which he called it as content boost collaborative filtering. Costin-Gabriel Chiru et al. proposed a system which recommends movies to the user based on the user data. It uses the user psychological profile and ratings of the films from different sites and users past ratings all these information is taken into consideration and recommends movies to the user. It solves the problem of unique movie recommendation to all the users [9].

III. PROPOSED METHODOLOGY

The Content based filtering is applied with the usage of better metadata which increases the quality of our recommender than using only overview column of our dataset to find recommendations. To overcome sparsity and scalability problem in collobarative filtering single value decomposition is proposed for better performance. Cosine similarity [10] is used in content based filtering which is efficient than euclidean distance. Three main strategies are used for our recommender system. One Demographic Filtering i.e. They offer general suggestions for each individual, based entirely on the film's image and / or genre. The program recommends similar films to users with the same census function. if you consider that each person is of the same type, this method is considered very simple. The logic behind this program is that movies that can be popular and critically acclaimed can have a high impact on ordinary audiences. The second method is content based filtering, in which we will take all the features like director, actors and movie related content and based on that we will recommend movies. The third one is collaborative filtering, in which we are implementing item based collaborative filtering and single value decomposition.

There are different types of movie recommendation systems are available with various methods.

1. Demographic Filtering
2. Content based Filtering
3. Collaborative Filtering

1. Demographic Filtering: It is a general method where it provides the general recommendations to the user based on the film demand. By applying the demographic filter to the system the system will recommends the one and the same film to the users with near characteristic [11]. When you consider that every end-user is distinct, this method is easy. In the demographic filtering the objective of the system is the to recommend the most popular and welcomed movies. So, that there is more possibility of being liked by the most of the people. To apply the Demographic filtering, we need to pass parameters like

- Measure to attain the score or rate movie
- Find the rating for each and every movie in the market.
- According to the scores that obtained provide the excellent scored films to the end_user.

By the mean rating of the film with score however mistreatment this may not be honest enough since a with 7.9 mean rating of only four votes cannot be taken into recommendation than the film with 6.8 mean rating even so with 40 votes. To overcome this, we can utilize the IMDB's Weighted rating principle as shown in given below.

$$\text{Weighted Rating (WR)} = \left(\frac{v}{v+m} \cdot R\right) + \left(\frac{m}{v+m} \cdot C\right)$$

Where v-number of votes of movie.

m-minimum number of votes required in list

R-average rating of movie

C-mean vote across whole report

2. Content-based Filtering: In this method, films are suggested based on the product profile and customer profile. The customer profile is the is fulfilled to set up and to applicable to the customer in formation of keywords. The customer profile force to be seen as a set of keywords taken by an algorithm from products applicable via the person [12]. The group of keywords of an product is called product profile. Consider an example someone is going to purchase their favorite object 'X'. Unluckily the object 'X' has offered disclosed and because the seller suggests to individual to purchase object 'y' that create elements much like object 'X'.

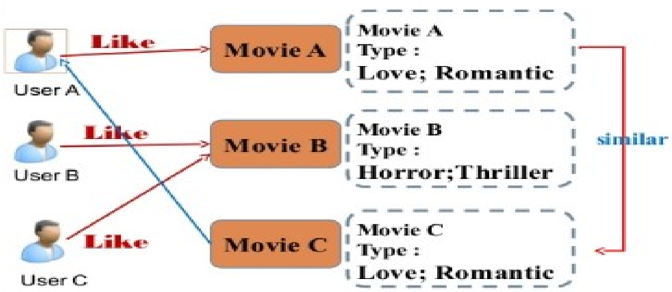


Fig. 1. Content Based Filtering Based Filtering

Here we are using cosine similarity to compute numeric quantity that indicates the likeliness among the films.In the same way we are using the cosine similarity to see the score where we can calculate easily and fastly.

The symbolic representation as follows

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}},$$

Now we can define the suggestion function. To define the suggestion funtion

we must follow some steps:

- Obtain the pointer of the film given in the title.
- Obtain the cosine similarity scores for specific film among all films.
- Change them into tuples where the first and second element place is likeliness score.
- Sort the tuple based on the scores.
- Obtain the ten elements from the top of the list. Don't consider the first element
- Go back to the titles corresponding to the positions of peak elements.

Our model carried out the fair task of getting the movies with homogeneous explanation the suggestions is not great. "The darkish Knight Raise" get all another film where it is most credible to the users that film is greater likely to enjoy different films. Something it cannot be taken by the way of the current system.

3. Collaborative Filtering: There are some limitations in the content-based filtering. It is able to suggest films which are on the brink of a particular film. It is unable to catch the flavor and give the recommendations across genres. The model that we built isn't catch the private tastes and partial to the end-user anyone ask our model for the recommendations [13] supported a film will get an equivalent recommendation for that film, no matter who is end user for that we will use collaborative filtering to provide movie recommendations.

It is divided into two types:

a) User based Filtering:

The model provides the recommendations to the customer that the same type customers had liked. To compute the likeliness among the end users we can use the cosine similarity and pearson correlation. This user-based filtering method can be explained with an example with help of the table. In the table each reindicates the end-user and columns represent to various films. The last row represents the likeliness among the end-user and target, The cells in in the table indicates the rating that is provided by the customer to the item.

	The Avengers	Sherlock	Transformers	Matrix	Titanic	Me Before You	Similarity(i, E)
A	2		2	4	5		NA
B	5		4			1	
C			5		2		
D		1		5		4	
E			4			2	1
F	4	5		1			NA

Fig. 2. User-based Filtering

Consider target as User E: The User A and User F don't have any kind of similarity with user E. So their similarities aren't defined in with Pearson Correlation. Take the user B, user C, and user D to build the Correlation and then calculate the subsequent similarity.

	The Avengers	Sherlock	Transformers	Matrix	Titanic	Me Before You	Similarity(i, E)
A	2		2	4	5		NA
B	5		4			1	0.87
C			5		2		1
D		1		5		4	-1
E			4			2	1
F	4	5		1			NA

To calculate user based collaborative filtering is easy but there are some limitations [14]. The end-user preference and taste changes continuously. It specifies that the pre calculation of the matrix based on their adjacent end-users which shows worst performance. To overcome from this issue item-based collaboration filtering is introduced.

b) Item based Collaborative Filtering:

Rather than considering the likeliness between the customers, this method provides the suggestion by considering the likeness that the films that the target customer rated. In the same way the Pearson correlation or Cosine Similarity is calculated. In the table each row indicates the end-user and columns represent to various films. The last row represents the Similarity. The Following table shows item-based filtering.

	The Avengers	Sherlock	Transformers	Matrix	Titanic	Me Before You
A	2		2	4	5	2.94*
B	5		4			1
C			5		2	2.48*
D		1		5		4
E			4			2
F	4	5		1		1.12*
Similarity	-1	-1	0.86	1	1	

Fig. 3. Item-based Filtering

To find the similarity between the items the below formula is used

$$sim(i,j) = \frac{\sum_{u\in U}(R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u\in U}(R_{u,i} - \bar{R}_i)^2}\sqrt{\sum_{u\in U}(R_{u,j} - \bar{R}_j)^2}}$$

where sim(i,j) is the similarity weighted value between the movies i and j and also R(u,i) is the rating given by the user ‘u’ to the movie ‘i’.The average rating of the movies given by user ‘u’ is represented by ‘Ru’.

The item based Collaborative Filtering [15] is static and it gives wide breath to the hassle posed through dynamic end-user preferences. There are several issues present in this technique. The major problem is extensibility. Calculation extends with the end-user and consequently item. The complexity is O(pq) in worst case where p is end-users and q are items. Another problem is sparsity. In the above table one customer rated both Titanic and Matrix, likeliness among them is one. In majority occurrences many numbers of end-users and therefore the likeliness between the two special films might extremely high just due to the fact they need similar rank by the single end-user who rated them both.

c) Single Value Decomposition: The sparsity and scalability problem created by the collaborative-based filtering can be handled by using the single value decomposition. It also reduces the time complexity that is caused by collaborative-based filtering. By reducing the dimensionality, it will reduce the number of computations and decreases the time complexity. we are converting this recommendation problem into optimization problem. It uses the latent factors for reducing the dimensionality. Each latent factor is mapped against to the user and the item in the dimensionality space ‘r’. By using this single value decomposition, we can also find Root Mean Square Error which is a metric to evaluate the performance of the system. If rmse value is less than our systemis working good.

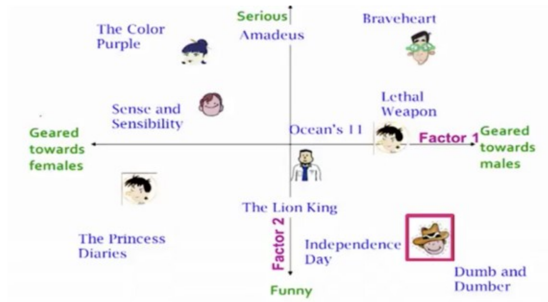


Fig. 4. Single value Decomposition

IV. RESULTS AND DISCUSSION

By applying the above condition 481 movies are qualified and these movies are send for further processing. Now we have to calculate the weighted rating for that we are defining a function called weighted_rating( ) in which it will calculate the metric and we are defining a feature called score it will store the results of the weighted_rating( ) function in our qualified dataframe. Now we are at final stage in which we will perform sorting to our dataframe based on the score value with which we will get the top score movies at the beginning. The top 15 movies with their title,score,vote_count,vote_average are displayed.

	title	vote_count	vote_average	score
1881	The Shawshank Redemption	8205	8.5	8.059258
662	Fight Club	9413	8.3	7.939256
65	The Dark Knight	12002	8.2	7.920020
3232	Pulp Fiction	8428	8.3	7.904645
96	Inception	13752	8.1	7.863239
3337	The Godfather	5893	8.4	7.851236
95	Interstellar	10867	8.1	7.809479
809	Forrest Gump	7927	8.2	7.803188
329	The Lord of the Rings: The Return of the King	8064	8.1	7.727243
1990	The Empire Strikes Back	5879	8.2	7.697884

Fig. 5. Dataset with attributes

We have made our first recommender. Now we got the top 10 movies which are more popular based on the weighted rating. It will recommend same movies to all the users.

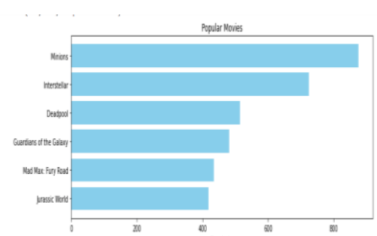
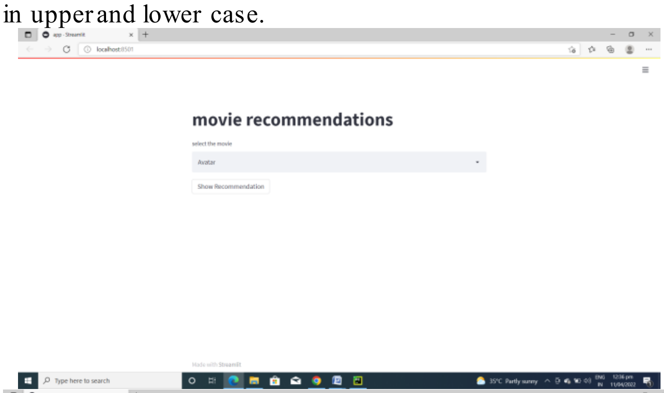


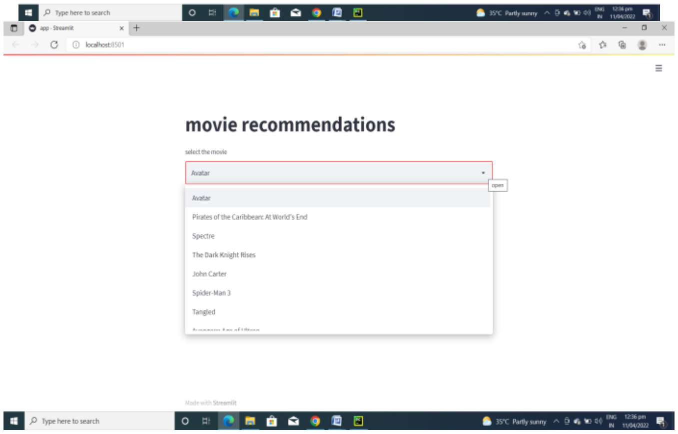
Fig. 6. Number of pupular movies versus total movies in dataset

From the above we have plotted those popular movies by taking popularity on x-axis and movies on y-axis. Something we have to remember is that thos demographic-filtering is not user specific and it will show results to all the users and it does not consider taste and preferences of the user. So this is not a better solution for movie recommendation, so we are moving to the next filtering i.e content-based filtering.

Credits, Genres and Keywords Based Recommender: It proceed without giving the instruction that the standard of our suggestionwould be exaggerated with the usage of more information. That is absolutely that getting to neutralize this segment.we are going to construct a recommender supported subsequent more information.The three top directors ,acotors the directors and related genres and therefore the movie plot keywords.from the solid crew and keyword options we'd prefer to extract the 3 most important actores [16], directors and thus the keywords related to the picture show.Right now our knowledge gift with within the type of stringified lists.we'd prefer to change into secure and usable structure.

In the next step we will remove spaces and convert all the columns values into lower case so that our count-vectorizer will not count the “johny” of “johny depp” and “johny walz” as same and also it does not count the same words which are in upper and lower case.





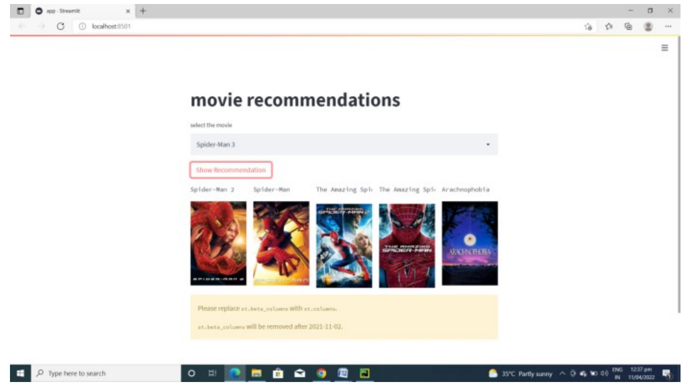


Fig. 7. Screenshots for movie recommendation system

RMSE (Root-Mean-Squared-Error) is applied as an evaluating criterion to analyze the performance of recommender system. If the value of RMSE value is lower, then the performance of the system is better. We got mean Root Mean Square Error [17] of 0.898 approx which is more than good enough in this case.

V. CONCLUSION

The Recommendation system what we have build will recommend movies to the user when he or she selects a movie. This is based on content based filtering in which it will consider the contents like director, actor, genres of the movie and recommend the similar movies. Next we had implemented collaborative filtering which is user specific it will predict the ratings of the movie by a particular user based on the related items and recommend movies based on similarity between the users or similarity between the items. At last we had implemented the single value decomposition to reduce the dimensionality that is occurred in the item based collaborative filtering.

In this proposed approach we had used contents like director,actors and genres in content based filtering but in future we can also consider age of the user as the preferences changes with age of the user. In future we can implement hybrid filtering which is a combination of content -based filtering and collaborative-based filtering in which it consider contents and similarities between user and item.Hybrid recommender will also reduce the value of Root mean square error which intum increase the performance of recommeder system and which increases precision and accuracy.We can also explore this recommendations to various domains like videos,songs,news,tourisms etc.

REFERENCES

[1] Sudhanshu Kumar , Kanjar De and Partha Pratim Roy “Movie Recommendation System Using Sentiment Analysis From Microblogging Data,”. IEEE Transactions on computational social systems., vol. 7, no. 4, pp. 915-923, Aug. 2020.

[2] Mukesh Kumar kharitha, Akul kumar,pardeep singh “Item based collaborative filtring in movie recommendation system in real time,” IEEE 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), doi:10.1109/ICSCCC.2018.8703362.

[3] G. Adomavicius and A. Tuzhilin, “Toward the next generation of recommender systems: A survey of the state-of-the-art and possible

extensions," IEEE Trans. Knowl. Data Eng., vol. 17, no. 6, pp. 734–749, Jun. 2005.

[4] O. Araque, I. Corcuera-Platas, J. F. Sánchez-Rada, and C. A. Iglesias, "Enhancing deep learning sentiment analysis with ensemble techniques in social applications," Expert Syst. Appl., vol. 77, pp. 236–246, Jul. 2017.

[5] E. Aslanian, M. Radmanesh, and M. Jalili, "Hybrid recommender systems based on content feature relationship," IEEE Trans. Ind. Informat., early access, Nov. 21, 2016, doi: 10.1109/TII.2016.2631138.

[6] J. Bobadilla, F. Ortega, A. Hernando, and J. Alcalá, "Improving collaborative filtering recommender system results and performance using genetic algorithms," Knowl.-Based Syst., vol. 24, no. 8, pp. 1310–1316, Dec. 2011

[7] Jun Wang , Stephen Robertson , Arjen P. Vries , Marcel J. Reinders,

[8] "Probabilistic relevance ranking for collaborative filtering", Information Retrieval, Volume 11, Issue 6, pp. 477-497, Springer, December 2008.

[9] Schafer J. B., Frankowski D., Herlocker J., and Sen S., "Collaborative filtering recommender systems," Lecture Notes In Computer Science, vol. 4321, pp. 291-324, 2007.

[10] Mohammad Yahya H. Al-Shamri , Kamal K. Bharadwaj; "A Compact User Model for Hybrid Movie Recommender System " International Conference on Computational Intelligence and Multimedia Applications, 2007.

[11] Christina Christakou, Leonidas Lefakis, Spyros Vrettos and Andreas Stafylopatis; "A Movie Recommender System Based on Semi-supervised Clustering ", IEEE Computer Society Washington, DC, USA, 2015.

[12] Mukherjee, Rajatish, Neelima Sajja, and Sandip Sen. "A movie recommendation system–an application of voting theory in user modeling." User Modeling and User-Adapted Interaction 13.1, pp. 5- 33, 2003.

[13] Ahn, Shinhyun, and Chung-Kon Shi. "Exploring movie recommendation system using cultural metadata." Transactions on Edutainment II, Springer, Berlin, Heidelberg, 2009. 119-134

[14] Miller, Bradley., et al. "MovieLens unplugged: experiences with an occasionally connected recommender system", Proceedings of the 8th international conference on Intelligent user interfaces, ACM, 2003

[15] Lekakos, George, and Petros Caravelas. "A hybrid approach for movie recommendation", Multimedia tools and applications36.1, pp. 55-70, 2008.

[16] Deng Ai-lin, Zhu Yang-yong and Shi Bo-le, "A collaborative filtering recommendation algorithm based on item rating prediction", Journal of Software, 2003

[17] Shyam Mohan, J.S., Vedantham, H., Vanam, V., Challa, N.P. (2021). Product Recommendation Systems Based on Customer Reviews Using Machine Learning Techniques. In: Jeena Jacob, I., Kolandapalayam Shanmugam, S., Piramuthu, S., Falkowski-Gilski, P. (eds) Data Intelligence and Cognitive Informatics. Algorithms for Intelligent Systems. Springer, Singapore.